

REMARKS/ARGUMENTS

Claims 1-54 are in the case.

The applicants have studied the office Action dated June 3, 2004, and believe the application is in condition for allowance. Reconsideration and reexamination are respectfully requested.

Claims 1-2, 4-6, 16-21, 23-29, 31-37, 39-41, 43-49, and 41-54 have been rejected under 35 U.S.C. 102(b) as being anticipated by U.S. Pat. No. 5,119,377 to Cobb et al., (Cobb). Claims 3, 7, 22, 38 and 42 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Cobb in view of the background section. Claims 15, 30 and 50 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Cobb in view of U.S. Pat. No. 6,708,291 to Kidder. These rejections are respectfully traversed.

Claim 1 is directed to a “method for handling errors in a service program ...” comprising, *inter alia*, “... performing a series of test operations when executing the service function call, wherein each test operation returns an identifier if the test operation failed, wherein each identifier is associated with one error state; and if one test operation failed, transmitting the identifier associated with the error state of the failed test operation to an error handling agent, wherein the error handling agent accesses error information associated with the identifier and generates error information describing a specific cause of the error that caused the test operation to fail.” It is the Examiner’s position that Cobb “discloses: ... performing a series of test operations when executing the service function call ... each identifier is associated with one error state (col 2, lines 48-68, col 3, lines 1-38, col 7, lines 1-5) ...” The applicants strongly disagree.

The Examiner’s citations to the Cobb reference do not teach or suggest “... performing a series of test operations when executing the service function call ...” as required by claim 1. Instead, it appears that, for example, the Cobb reference states:

When this logic detects an error, the process is called and collects only the data required to diagnose the error. This process is known as the Early Detection Data Capture (EDDC) process. It is a table-driven data collection and reporting routine which captures very specific program information, identified at the point of error detection. It builds a unique problem identifier, known as a software symptom string, and sends an automatic

notification of the problem to an operator as is done, for example, using the IBM System Network Architecture generic alert function in a computer network utilizing this architecture. Cobb, col. 3, lines 6-18.

Thus, the Examiner has cited no portion of the Cobb reference which describes the performance of a series of test operations when the Cobb process is called in response to the detection of an error in the Cobb system.

Moreover, the Examiner's citations to the Cobb reference do not teach or suggest execution of a service function call in which "each test operation returns an identifier if the test operation failed, wherein each identifier is associated with one error state ..." as required by claim 1. The above cited description of the EDDC process of the Cobb reference merely describes the return of "a problem identifier" by the EDDC process called in response to the detection of an error.

Thus, not only has the Examiner cited no portion of the Cobb reference which describes the performance of a series of test operations when the Cobb process is called in response to the detection of an error in the Cobb system but the Examiner has cited no portion of the Cobb reference which describes each such test operation of a series of test operations returning an identifier if the test operation failed, wherein each identifier is associated with one error state. Lacking a description of such a series of test operations, it is of course also clear that the Examiner's citations to the Cobb reference do not teach or suggest execution of a service function call in which "if one test operation failed, transmitting the identifier associated with the error state of the failed test operation to an error handling agent, wherein the error handling agent accesses error information associated with the identifier and generates error information describing a specific cause of the error that caused the test operation to fail" as required by claim 1. Independent claims 5, 20, 36 and 40 may be distinguished in a similar fashion. Moreover, the dependent claims include additional limitations, which in combination with the base and intervening claims from which they depend provide still further grounds of patentability over the cited art.

The Examiner's citations to the Kidder reference do not obviate the deficiencies of the Examiner's citations to the Cobb reference. It is therefore respectfully submitted that the rejections of claims 1-54 should be withdrawn.

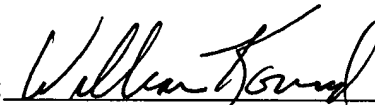
The Examiner has made various comments concerning the anticipation or obviousness of certain features of the present inventions. Applicants respectfully disagree. Applicants have addressed those comments directly hereinabove or the Examiner's comments are deemed moot in view of the above response.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-54 are patentable over the art of record. Applicants have not added any claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0449.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: September 3, 2004

By: 

William K. Konrad
Registration No. 28,868

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984